REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Washington Headquarters Service, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington, DC 20503.

PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.

1. REPORT DATE *(DD-MM-YYYY)* SEP 2011	**2. REPORT TYPE** Conference Paper (Post Print)	**3. DATES COVERED** *(From - To)*

4. TITLE AND SUBTITLE

AFFORDABLE EMERGING COMPUTER HARDWARE FOR NEUROMORPHIC COMPUTING APPLICATIONS

5a. CONTRACT NUMBER
IN HOUSE

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Morgan Bishop (AFRL), Michael J. Moore (ITT), Daniel J. Burns (AFRL), Robinson Pino (AFRL), Richard Linderman (AFRL)

5d. PROJECT NUMBER
23T1

5e. TASK NUMBER
PR

5f. WORK UNIT NUMBER
OJ

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

AFRL/RITC
525 Brooks Road
Rome, NY 13441-4505

8. PERFORMING ORGANIZATION REPORT NUMBER

N/A

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

AFRL/RITC
525 Brooks Road
Rome NY 13441-4505

10. SPONSOR/MONITOR'S ACRONYM(S)
N/A

11. SPONSORING/MONITORING AGENCY REPORT NUMBER
AFRL-RI-RS-TP-2011-37

12. DISTRIBUTION AVAILABILITY STATEMENT

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED. PA #: 88ABW-20100-0292
DATE CLEARED: 28 JAN 2010

13. SUPPLEMENTARY NOTES

Publication in The 2010 International Joint Conference on Neural Networks (IJCNN), 18-23 July 2010, pp 1-5, Barcelona, Spain, ISSN: 1098-7576, Print ISBN: 978-1-4244-6916-1, INSPEC Accession No.: 11593907, Digital Object Identifier: 10.1109 / IJCNN.2010.5596664 Date of Current Version: 14 Oct 2010. One or more of the authors is a U.S. Government employee working within the scope of their Government job; therefore, the U.S. Government is joint owner of the work and has the right to copy, distribute, and use the work.

14. ABSTRACT

We are pursuing an investigation of neuromorphic computational models and architectures in order to leverage present understanding of how the estimated 10^{11} neurons and 10^{15} neuron connections in the mammalian brain are able to do some of the things a human does, and as quickly as it does it, using slow base components, while consuming very little power on affordable synthetic non-biological computing hardware. Understanding and harvesting neurologically based methods is a promising approach with great potential that may help us achieve massively parallel computation far beyond the scope of traditional computing.

15. SUBJECT TERMS

Neuromorphic, Cognitive, Computing, Memory, Emerging Technology, Computational Intelligence

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
					ROBINSON E. PINO
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			**19b. TELEPHONE NUMBER (*Include area code*)**
U	U	U	UU	6	N/A

Affordable Emerging Computer Hardware for Neuromorphic Computing Applications

Morgan Bishop, Michael J. Moore, Daniel J. Burns, Robinson E. Pino, *Senior Member*, *IEEE*, and Richard Linderman, *Fellow*, *IEEE*

Abstract—**We are pursuing an investigation of neuromorphic computational models and architectures in order to leverage present understanding of how the estimated 10^{11} neurons and 10^{15} neuron connections in the mammalian brain are able to do some of the things a human does, and as quickly as it does it, using slow base components, while consuming very little power on affordable synthetic non-biological computing hardware. Understanding and harvesting neurologically based methods is a promising approach with great potential that may help us achieve massively parallel computation far beyond the scope of traditional computing.**

I. Introduction

THIS effort has explored the issues associated with the efficient mapping of neuromorphic computing strategies onto advanced computational architectures. The computing performed by neurological systems produces cognitive phenomena that have been high value, yet elusive, goals of computational researchers. Neuromorphic computing, as evident in primate brains, uses massive collections of modest speed synapses and neurons operating asynchronously in parallel.

It is becoming feasible to emulate full scale brains on a neuron level, at least insofar as computational complexity matters. The human brain has an estimated 10^{11} neurons, each with an average estimated 10^4 connections to other neurons. Single neuron models need to account for synapses (connections) and somas (cell bodies). A simple synapse model uses two numerical operations (OPs): an index (address addition) and a value addition (this would be the complexity floor). A simple soma model (threshold compare and assignment) is equivalent to two OPs. Thus, human brain emulation (if all neurons and synapses happen to fire at once; an unlikely event) would require ~3X10^{15} OPs. A single Cell Broadband Engine® (Cell-BE) chip can peak at 2X10^{11} FLOPS. 15K such devices, by this measure, would be able to emulate a full sized human brain at about 1/1000 real-time speed. Certainly, synapse and neuron level models can be more complex than this estimate, but it is also true that emulation may not always need to be carried out at a low level. Moreover, it is often the case that one neuron connects with another multiple times, a situation that can be simplified in emulation by allowing for a "wider" weight range.

We explored multiple columnar cortical models reported in the literature, and produced new models by combining ideas with insights developed by the team. These models range in scale of abstraction from cell assemblies of individual minicolumns to models that represent abstractions of hundreds of thousands of synapses and neurons. In each case, effort was made to understand neuron-based computational underpinnings, the cognitive efficacy of the model, the fit of the digital emulation of the model to computer architectural features, and the scaling of the model into a full-scale system. Selected models were also simulated.

II. Experimental details

A. Neuromorphic Primary Visual Cortex (V1) Model

An estimate of computational complexity of full scale V1 emulation was made to look at the feasibility of full scale modeling of cortical fields. The estimate was based on representing minicolumns as "integrate and fire neuron" models. This kind of neuron scale emulation is thought to be more computationally demanding than more abstract, less neuron based models, and thus serve as a conservative estimate. However, it is far simpler than a spiking dynamical model and, as it stands, does not account for many dynamical characteristics of neurons.

The "integrate and fire" neuron model involves the summation over the synapses of a neuron, and is then subjected to a threshold function. The synapse summation is a weighed summation, equivalent to a dot product between a weight vector and a neuron value vector. There are about 180 neurons in a V1 minicolumn, and perhaps 30 of them are tightly recurrent within a minicolumn. These would be connected to fewer neurons than the others and we placed that estimate at 100. The other 150 neurons are assumed to be connected to about 1000 neurons.

Manuscript received January 15, 2010. This work was supported by the U.S. Air Force Office of Scientific Research under LRIR # 061F02COR. This paper represents a more concise summary of that work's final technical report.

M. Bishop is with USAF AFMC AFRL/RITC, 525 Brooks Road Rome, NY 13441-4505 (phone: 315-330-1556; fax: 315-330-2953; Morgan.Bishop@rl.af.mil).

M. J. Moore is with ITT/AES, 775 Dandelion Drive, Rome NY 13441 (phone: 315-330-1500; e-mail: Mike.Moore@itt.com).

D. Burns is with USAF AFMC AFRL/RITC (Emeritus), 525 Brooks Road Rome, NY 13441-4505

R. E. Pino is with USAF AFMC AFRL/RITC, 525 Brooks Road Rome, NY 13441-4505 (phone: 315-330-7109; fax 315-330-2953; Robinson.Pino@rl.af.mil).

R. Linderman is with USAF AFMC AFRL/RI, 525 Brooks Road Rome, NY 13441-4505 (phone: 315-330-2208; fax 315-330-2953; Richard.Linderman@rl.af.mil).

Dot product complexity:

2N (N multiply, or Add operations)
Total = 2N (1000) + 2M (100), N = 150, M = 30
= 306,000 FLOPs.

To accommodate communication between minicolumns, we make the assumption that these neurons would cycle typically 5 times per saccade (a saccade is a rapid eye movement):

5X306,000 = 1,530,000 FLOPs.

There are about 5 saccades/second:

7,650,000 FLOPS/Minicolumn/second

There are about 1.6 million minicolumns/V1:

1,600,000 X 7,650,000 = 12.16 TFLOPS,

V1 is about 1% of the entire neocortex, but the neuron density in V1, based on minicolumn characteristics, is about twice the density beyond V1. A rough estimate of whole neocortex complexity is:

50X12.16 TFLOPS (608 TFLOPS)

B. High Performance Computing

The previous rapid rate of clock speed increase for CPUs has disappeared. Chip developers have turned to multicore technology to make use of the continuing exponential trend towards increased transistor density. Multicore technology shifts problems from hardware to software and multiplies available parallelism. To make productive use of 100 thousand to 1 million processors, one must provide software, which can efficiently harness the parallelism inherent in the hardware. Software development is labor intense. The cost grows significantly as parallelism increases. Software developers have few methods available to them to deal with parallel system design, except for messaging systems and multithread programming. No significantly better methods have emerged into common practice which displace or build upon these. These techniques are suitable for small scale parallelism but grow unwieldy for systems of even a few thousand processors. Existing High Performance Computer (HPC) platforms, like Blue Gene/L, can be configured with more than 130K processor cores. The challenge of harnessing parallelism on that scale for all but an "embarrassingly parallel" application (an application where very little communication is needed between processes) challenges the limits of programmability. Yet neural processing effectively harnesses parallelism on at least this scale.

Cognition presents as an excellent target of study because primate brains are examples of the kind of computing architecture we seek. It also holds promise to meet the "programmability challenge" of large scale parallelism with self supervised learning, and is therefore itself potentially a key technology for approaching other difficult to scale applications like Parallel Discrete Event Simulation (PDES). PDES applications are models of physical processes in terms of state changes at discrete points in time. These applications are characteristically intense in terms of CPU but challenge computer architectures with the need to communicate events to all affected elements within the simulation.

This effort has produced some infrastructure suitable for continuing cortical modeling research. It consists of software, in addition to the models discussed, developed for and applied to modeling a visual input stream (a retina model, an optic chiasm model, and a thalamic-LGN model), a high throughput Publish/Subscribe messaging system, and high performance machine clusters (288 SONY PlayStation3® Cell-BE platforms with 12 Dual Quad 3.2 GHz Xeon head nodes).

Each PS3 Cell-BE node has a Power PC core (PPE) and six satellite broad band engines (SPE). SPEs have small memories: 256K bytes, but can process floating point data rapidly (25.5 GFLOPs). Two high speed DMA channels (in, out) connect each SPE to a PPE. The PPE runs LINUX and has IP communication with the XEON head nodes and other PS3 nodes.

C. Brain State-in-a-Box

The Brain State-in-a-Box (BSB) algorithm was selected as the attractor function to incorporate into the network study because of its association with the Ersatz Brain project [1]. Ersatz Brain is an effort to model aspects of mind with nested networks of fixed point attractors. BSB uses state vectors with "N" real numbers in the range of (-1.0...+1.0). Its name is a metaphor for describing the algorithm as an N dimensional shape. Its fixed basin points of attraction lie in its corners. An N dimensional BSB function can separate M basin points, where M is ~15% of N. The model has many applications including machine reading, author ID, and scene interpretation. Applying the model efficiently involves exploring architecture design space, implementations, and evaluations of neuromorphic computing models. Preliminary assessment of the attractors suggested these attractors were useful for recognizing features using feed forward (afferent) data as well as feedback (expectation) data.

Details of implementing a 128-dimensional BSB model on the Cell processor can be found in [2, 3]. Referring to Figure 1, in the large-scale BSB model implementation, 128-dimentional BSB models are run on each of the six Synergistic Processing Elements (SPEs) on the Cell processor. The data communication functions are implemented on the PowerPC Processing Element (PPE), and the word and sentence level confabulation models are implemented on cluster head nodes associated with groups of 24 Cell-BE nodes. The BSB model was also implemented in an FPGA hardware version that achieved ~150 speedup over software [3, 4].

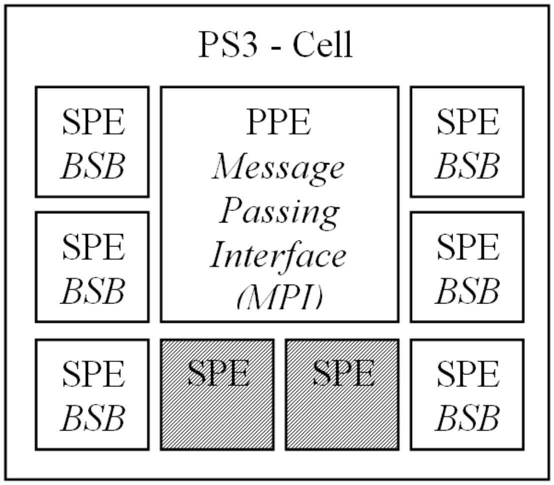

Fig. 1. Task distribution of one PS3 node.

TABLE I
PERFORMANCE, POWER, COMMUNICATION AND MODELING CAPABILITIES: 1 PS3 VS. 288

	1 PS3	288 PS3
Peak computational performance achievable by BSB application	102 GFLOPS	29.4 TFLOPS
Number of 128-dimensional BSB models supported (10ms reaction time)	3,000	864,000
Number of mini-columns from the visual cortex of the human brain modeled	12,000	3,456,000*
Total power consumption	140 W	40 KW
Achieved total network bandwidth for the communication test using MPI	~1Gb/s	~12Gb/s

*The V1 layer of the visual cortex consists of about 1,600,000 minicolumns.

Table 1 shows a comparison of the computing performance, communication performance, power consumption, and modeling capabilities between a single PS3 (1 Cell-BE with 6 SPEs), and the whole cluster (288 PS3s). Theoretically, we can implement two V1 layers of the human visual cortex on this cluster.

D. *Confabulation Model*

An investigation of confabulation surfaced reports by Robert Hecht-Nielson of a cognitive mechanism which explains all of cognition [5]. The center piece of his reports both published and in presentations, was a demonstration of software which completed sentences with no context, and another which completed a sentence in the context of two other sentences. The hypothesis is that the reported algorithm models the fundamental cognitive mechanism, and that the mechanism must be somehow layered on a large scale (many interconnected confabulators) to produce a level of coherence. The algorithm is computationally similar to Bayesian Belief, but it does not use a Belief tree network. It was decided to explore Confabulation first in its reported context (textual data) and to consider it later on as a candidate for extra striate (above V1) modeling, fulfilling an expectation role.

The reported sentence completion algorithm trained by reading text; lots of text. It then “recalled” by using a context (for example, the start of a sentence) to retrieve a sequence of words and phrases which its training statistically connected to the context. The training consisted of reading one sentence at a time and breaking it into sequences of words and phrases - all possible combinations of these. Sentence by sentence the training keeps track of all words and phrases encountered, and all sequences formed, through statistical links.

E. *Publish/Subscribe Communication*

A Publish/Subscribe (Pub/Sub) messaging model provides a very flexible method of system configuration without having to attend to details of physical node availability and node inclusion or exclusion. The system middleware used for this, a version of JBI developed at AFRL/RI, performed well within efficiency needs.

We examined the ability of the Pub/Sub communication model to distribute visual data pieces over a large set of processes. With visualization and the Pub/Sub server running on a dual quad platform with one retinal model, the chiasm process and one LGN process, a single subfield process was able to execute at about 2 frames per second. Real-time is probably 5 frames per second, corresponding to 5 saccades.

III. RESULTS AND DISCUSSION

The use of IBM Cell-BE technology (Sony PlayStation® 3 platform) to accelerate BSB performance was investigated. Runtime measurements show that we have been able to achieve about 70% of the theoretical peak performance of the processor when implementing a 128 element vector using a matrix shuffle strategy to improve Cell-BE SPE instruction utilization [6].

The 128 element BSB recall algorithm was implemented on a single SPE element of the Cell-BE architecture. The complexity is 33,280 FLOPs/ recursive cycle. Ten cycles are needed for convergence yielding 332,800 FLOPs/ recall. Peak efficiency corresponds to all floating operations being performed as quad word operations, with all other (non-floating point) instructions executing in the parallel instruction pipe. In this case, peak is 332,800/4 = 83,200 Quad Floating ticks. Each recall needs a weight vector load, a state vector load and a state vector unload (66,560 bytes) equivalent to 4160 quad word transfers (one quad word per tick). Compute to DMA peak ratio is therefore 83200/4160 = 20. Double buffering was used to overlap data transfer of weight matrices and state vectors with processing. Six BSBs can be run in parallel on a PS3 version of the platform. Efficient implementation on an SPE requires careful attention to aligning data for maximum effectiveness of intrinsic functions. Loop unrolling is essential as well to maintain the dual pipeline SIMD efficiency.

The 32 element BSB recall algorithm performs about 2240 floating operations for each recursive cycle; 2,176 for the actual algorithm and 64 for state vector conversions from and to integer fixed point. About 5 cycles are needed for convergence, yielding 11200 operations per 128 bytes of DMA data movement (no weight vector movement, and the state vector is actually 2 byte fixed point). Peak FLOP rate is (2176/4 + 64) 608 Quad Floating ticks/cycle. The peak DMA rate is (128/16) 8 DMA ticks. The peak compute to DMA ratio is therefore 608/8 = 76.

About 17 GFLOPs/Second (GFLOPS) were measured for the 128 element case. This corresponds to about 51,000 10 cycle recalls per second about 1/10th the rate achieved using the FPGA. However, six of these can be run in parallel on a single PS3 node chip, bringing the throughput to about half of the FPGA case. The Cell chip is more than an order of magnitude less expensive than the FPGA chip, and the Cell chip is programmed in C, compared to VHDL needed for the FPGA. By these measures the Cell technology has significant cost advantages over the FPGA technology.

A trial was run using all 288 PS3 notes in the Cell-BE cluster. The mark of 29.376 trillion FLOPS was reached.

About 11 GFLOPS were measured for the 32 element case. This corresponds to about 982,142 5 cycle recalls per second, about 1.5X faster than the tested FPGA doing the same work. However, since six of these can be performed in parallel in a PS3 node, the PS3 chip is potentially 9X faster than the FPGA.

Note that the 60 fold clock speed ratio (FPGA 100 MHz vs. Cell-BE SPE 6GHz) is a major factor in speed differences.

We researched, implemented, and evaluated the performance of the confabulation model, focusing specifically on two example application problems that we call here sentence completion and intelligent on-line character recognition (OCR). In both of these applications the basic problem is to complete a partial natural language sentence in a plausible, sensible way, given that only a fragment of the input sentence is available, and given that the system has been trained by exposure to a large training corpus of textual electronic media (e.g. books and news feeds). Good solutions to the sentence completion problem could very well translate to other input modalities (i.e. audio and imagery), and map to solutions in several higher level application scenarios.

We also spent some time looking at ways to speed up and scale up confabulation training and recall. The algorithms are ideal candidates for parallel processing and their performance can be significantly improved with the help of application specific, massively parallel computing platforms. However, as the complexity and parallelism of the hardware increases, the design effort and implementation costs also increase. Architectures with different cost-performance tradeoffs were analyzed and compared in [7], which describes hardware designs that achieved ~1,000x speedup of the confabulation training algorithm, and ~3,000x speedup of the recall algorithm. Our analysis showed that although increasing the number of field programmable gate array (FPGA) processing elements (PEs) or the size of memories per processing element can increase performance, the hardware cost and performance improvements do not always exhibit linear relationships. Hardware configuration options must be carefully evaluated in order to achieve good cost performance tradeoffs.

Three strategies were explored for optimization of the sentence completion algorithm: software optimization, software analysis and hardware architecture augmentation. Our analysis shows there is potential to improve the three structure techniques using hashing strategies. The hashing strategies may improve data locality as well. A hash version of training was demonstrated in about 4 seconds, compared to the 45 seconds the tree structures used. The cogent confabulation algorithm is an ideal candidate for parallel processing. It also shows that although increasing the number of processors or the size of memories can increase the performance of training and recall, the relations between resource cost and performance associated with these variations are not always linear. The details of hardware configuration must be carefully considered to achieve good cost performance tradeoffs. We suggest that this work can be extended to more complex implementations of confabulation systems.

IV. Conclusion

Neuromorphic computational architecture development is a new and accelerating field with significant promise. Individual qualifications to contribute in this domain include familiarity in multiple disciplines such as: computer architecture/technology, parallel software development, dynamical systems, neuroscience, neurology, neuropsychology, and agent based expert systems.

The results suggest topographically organized cortex, like "early" vision, audition and tactile sensing, can be emulated using minicolumn models similar to the hybrid model we created, and that the emulation is computationally tractable on, for example, a small number (hundreds) of Cell Broadband Engine® (Cell-BE) class chips. "Higher" cortical regions, because of plasticity needs, may require more computationally intense models, which deal with spiking dynamics and liquid state machine effects.

V. Future Work

We are in the process of procuring additional PS3 systems to increase the total number of PS3's to 2,016. The configuration will consist of 84 subclusters of 24 PS3's per subcluster. Each of the 84 head nodes will also have 2 GPGPU's; one NVIDIA Tesla C1060 and one NVIDIA Tesla C2050 for a total of 168 GPGPU's. Head node candidates are still being evaluated, but by combining computational power of all other processing components the cluster will have theoretical throughput of ~500 TFLOPS or ~.5 PFLOPS. The low price/performance ratio of the PS3 will allow for the creation of this system for less than $2M.

We estimate that this system will allow for the emulation of ~80% of the neocortex.

ACKNOWLEDGMENT

This work was supported in part by the United States Air Force Office of Scientific Research (AFOSR), LRIR# 061F02COR.

REFERENCES

[1] Anderson, J.A. (1993). The BSB network. Pp. 77-103 in MH Hassoun (Ed.), Associative Neural Networks, New York, NY: Oxford University Press.

[2] Wu, Qing; Mukre, Prakash; Linderman, Richard; Renz, Tom; Burns, Daniel; Moore, Michael; Qiu, Qinru; Performance Optimization for Pattern Recognition Using Associative Neural Memory. IEEE International Conference on Multimedia and Expo, 2008. On pages: 1-4. Publication Date: June 23 2008-April 26 2008.

[3] Richard Linderman. Qing Wu, Qinru Qiu, "FPGA and Cell Processor Performance Optimization for Brain-State-in-a Box (BSB) cognitive Computing", 2007 ARCS Symposium on Multicore and New Processing Technologies, Aug 2007.

[4] Qing Wu, Qinru Qiu, Richard Linderman, Daniel Burns, Michael Moore, Dennis Fitzgerald. "Architectural Design and Complexity Analysis of Large-Scale Cortical Simulation on a Hybrid Computing Platform." IEEE Computational Intelligence for Security and defense Applications (CISDA), 2007.

[5] Hecht-Nielsen R., Mechanism of Cognition. In: Bar-Cohen, Y. [Ed.] Biomimetics: Biologically Inspired Technologies, CRC Press, Boca Raton, FL (2006).

[6] Wu, Qing; Mukre, Prakash; Linderman, Richard; Renz, Tom; Burns, Daniel; Moore, Michael; Qiu,Qinru; Performance Optimization for Pattern Recognition Using Associative Neural Memory. IEEE International Conference on Multimedia and Expo, 2008. On pages: 1-4. Publication Date: June 23 2008-April 26 2008.

[7] Qinru Qiu, Daniel Burns, Michael Moore, Richard Linderman, Thomas Renz, Qing Wu. Accelerating Cogent Confabulation: an Exploration in the Architecture Design Space. 2008 Intl Joint Conference on Neural Networks, (IJCNN) at the 2008 IEEE World Congress on Computational Intelligence (WCCI).